

## ターゲット

今回のアプリのユーザのターゲットは、「今、流行りのフォトジェニックな写真を撮ることや、見ることが好き」と考えている人である。

## 便利な点

このユーザにとって、提示している多くの写真からユーザ好みの風景などの写真を素早く選べることが便利であると考えた。

ユーザに、現在地を基に現在地周辺で投稿されている写真を明示的に表示することで、ユーザが観光に行った際などで、観光スポットを決める一つの要因となり、手助けになると考えた。

また、都市の検索機能を設けることで、行きたい観光地を決めることができ、便利であると考えた。

## アプリ構成

今回のアプリでは、MVVM モデルで設計し、3 画面で構成している。また、今回は Flickr の API を使用した。

1 番目（図 1）は、アプリをスタートする時のスタート画面である。UIBlurEffect を用いて、背後の画像をぼやかして表示している。

2 番目（図 2）は、メイン画面である。UICollectionView を用いており、画面上部にボタンを 2 個設けている。また、検索のために、UISearchBar を用いている。`Near` を押すと現在地を基に現在地周辺で投稿されている写真を取得して再表示している。また、`Find` を押すと UISearchBar が現れ、テキストを入力できる（図 3）。そして、検索した時にその都市名から座標を取得してその座標周辺で投稿された写真を取得して再表示している。また、UICollectionViewCell に表示されている写真をタップすると、その写真の画像を取得して 3 番目の画面に値を渡して遷移している。

3 番目（図 4）は、メイン画面で UICollectionViewCell を選択した時に画面遷移して表示される画面である。この画面では、前画面で選択された写真の座標を基に、その座標周辺の写真を取得して表示している画面である。画面下部には、UICollectionView を用いて写真を表示している。



図 1. スタート画面

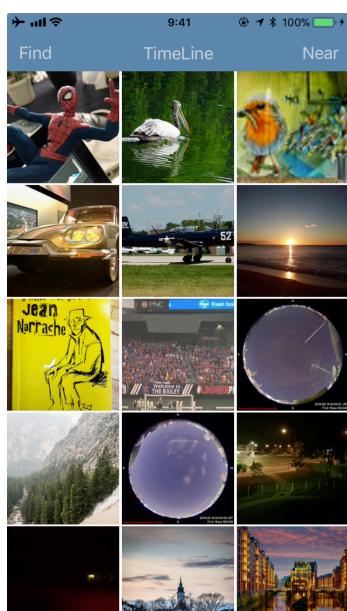


図 2. メイン画面



図 3. SearchBar 表示

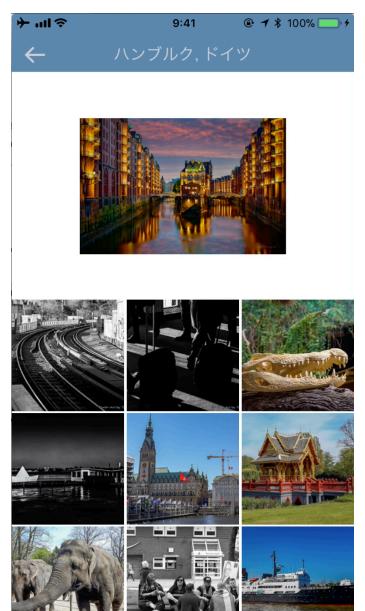


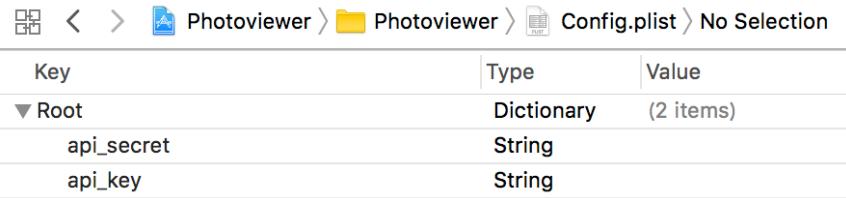
図 4. 都市周辺画面

## 手順

手順 1 では、`pod install` する。

手順 2 では、Flickr に登録し、api\_key と api\_secret を取得する。

手順 3 では、"Config.plist" (図 5) の api\_key と api\_secret の value に登録した自分の api\_key と api\_secret を入力する。



Key	Type	Value
Root	Dictionary	(2 items)
api_secret	String	
api_key	String	

図 5. Config.plist

## 今後の発展

実装開始の段階では、RxSwift と RxCocoa を用いて開発するつもりだったが、実装の規模感と実装日数を考えると今回は用いらない決断をした。今回は、クロージャーを用いて開発した。しかし、RxSwift を用いると非同期処理のコールバックの処理をシンプルに書くことができる。そのため、今後の発展として RxSwift に書き換える。

## 参考デモ

<https://github.com/tomoyamatsuyama/Demo/blob/master/demo1.gif>

作成日：2月 18 日

氏名： 松山友也